Adaptive Near Sensor Compressing for Energy Savings in Wireless Body Area Sensor Networks

Corentin Lavaud, Antoine Courtay, Matthieu Gautier, Olivier Berder

Adaptive Near Sensor Compressing for Energy Savings in Wireless Body Area Sensor Networks

Corentin Lavaud, Antoine Courtay, Matthieu Gautier, Olivier Berder
Univ Rennes, CNRS, IRISA
firstname.lastname@irisa.fr

Abstract

Wireless body area networks are essentially constrained by the energy required to send data from one node to another. While most of times the whole data is sent, data compression can be used to decrease the amount of data and therefore the energy consumption. However, the compression is energy consuming and its efficiency depends on data size and data type. To tackle this challenge, the proposed Adaptive Near Sensor Compressing (ANSC) algorithm performs pertinent feature extraction on the sensed data, and selects the best compressor by estimating the energy required by the hardware platform to perform both data compression and transmission. The transmission power is also dynamically tuned to reach a specific quality of service according to node location and propagation channel. ANSC has been evaluated on a real platform dedicated to wireless body area network and has shown an energy gain of up to 49% compared to full transmission.

1 Introduction

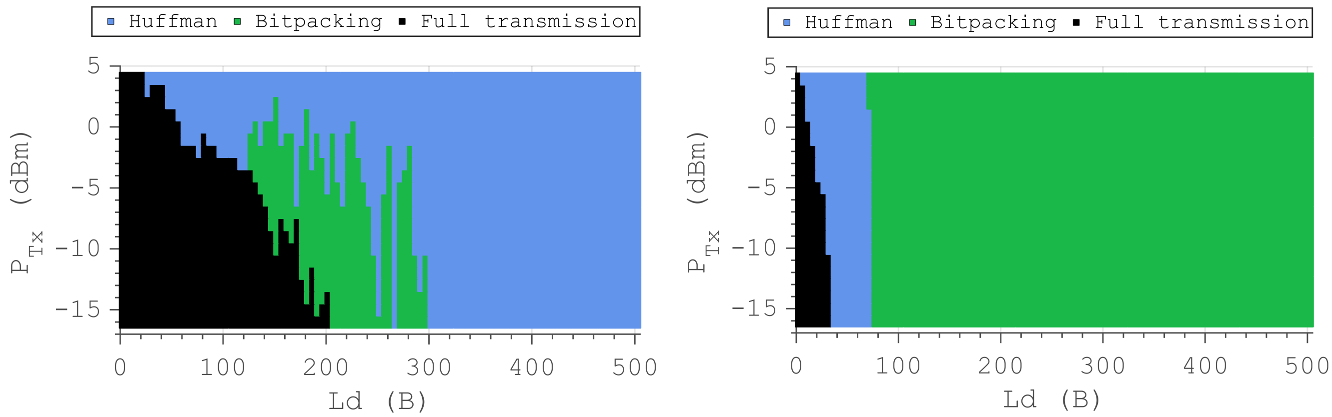
Wireless Body Area Network (WBANs) are commonly used for numerous application fields (medicine, security, sport science ...). They are composed of distributed, wearable and autonomous nodes that are able to sense on-body data and relay them using wireless communication to a base station known as the sink [2]. Nodes are typically energetically constrained and the optimization of their power consumption is critical to ensure a long lifetime. In most cases, the most consuming part of a node is the radio communication [10] and many efforts deal with methods allowing power consumption reduction. Most common techniques for saving energy are duty-cycling [7] which consists in using its sleep mode when a component is not used ; protocol optimization to reduce the number of control data exchanges [4] ; and also transmission (Tx) power adaptation [3] to send a data packet at the lowest power as possible without altering the transmission quality.

Another approach is to lower the amount of data to be sent by exploiting the processing capabilities of each node. To this aim, compression techniques could be used in order to reduce the data size and thus the radio energy consumption. Data compression is not so commonly used in WBANs [12]. The main reason is the limited hardware resources of a node. To achieve low power, both internal memory size and computation capabilities are small and compression algorithms naturally have to be adapted to the hardware constraints, such as S-LZW (Lempel-Ziv-Welch for Sensors) [11] and MiniLZO [1]. Moreover the size of the data to be compressed is often very small (less than 100 B) in WBAN applications, whereas most compression algorithms have been designed for a large amount of data.

In WBANs, the energy efficiency of compressors will depend on both the data type and the amount of data. Adaptive compressors are therefore a solution to deal with these time-varying parameters. OAC (Online Adaptive Compression) [6] switches between a LZW compressor and no compression depending on the data length, but it does not take into account the energy involved in all these steps. [13] studies the impact of the compression on the energy but relies on lossy compression techniques that are not always suitable to WBANs, especially in medical field or others requiring high precision.

In this paper, the combination of adaptive Tx power and compression is achieved by the Adaptive Near Sensor Compressing (ANSC) algorithm. ANSC is an energy-aware algorithm which uses the power characteristics of the hardware platform, the characteristics (length and type) of the data to be sent, and the power required to sent a frame to automatically switch between different compression schemes and transmission powers in order to reduce the overall energy used. Two compression schemes are considered, namely Huffman [8] and Bitpacking [9] compressors, and compared to full data transmission case. These compressors are chosen because they are both fast, require few memory and are lossless.

The rest of this paper is organized as follows. Section 2 introduces the energy model of both compression and transmission processes that will be used in Section 3, dedicated to the ANSC algorithm. In Section 4, ANSC performance evaluation on a WBAN platform is proposed, before conclusions



(a) Map with high variance dataset (`Zyggie moving` dataset). **(b)** Map with low variance dataset (`Zyggie idle` dataset).

Figure 1. Compressor energy efficiency maps.

are given in Section 5.

2 Energy models

The energy of a wireless sensor node is spent by both hardware and software components: hardware components are usually the radio transceiver, the processing unit and the sensors while the software components are, in this study, the compressors that are running on the processing unit. The main feature of our ANSC algorithm is its prediction of the compression behavior in terms of energy. Indeed, the compression efficiency depends on several factors such as the data characteristics and the amount of data. To choose the most energy-efficient compressors, accurate modeling of this compression efficiency is preliminary required.

2.1 Compressor model

Predicting the behavior of compression algorithms is a very challenging task. In order to be fast and thus be more energy efficient, the prediction of the compression ratio and the compression time are approximated using pertinent feature extraction. Curve fittings on microbenchmarks are used to generate models of the compressor behaviors. The data used in the various microbenchmarks are issued from WBAN dataset.

The energy used by a compressor is:

$$E_{\text{compression}} = P_{\text{CPU}} \cdot T_c, \tag{1}$$

with P_{CPU} the power of the processing unit and T_c the time needed to compress the data, which varies depending on the data and the compressor used.

The processing unit power consumption P_{CPU} is supposed to be constant and is measured when the processing unit was ongoing floating point computation. In fact, our measurements have shown that in an idle state (not sleep mode) or ongoing floating point computation results in nearly the same power consumption for the used target (ARM Cortex-M4). The real power consumption variation occurs when switching to sleep mode.

2.2 Hardware model

The estimation of the energy cost of a radio transmission requires to know the radio Tx power, the frame length and a model of the radio power. This last part is hardware specific and measurement is compulsory to create an accurate model:

$$E_{\text{radio}} = P_{\text{send}} \cdot T_{\text{send}} = f_1(P_{\text{Tx}}) \cdot f_2(Ld, br), \tag{2}$$

where $f_1()$ and $f_2()$ are two functions that respectively give the real radio power consumption P_{send} as a function of the Tx power P_{Tx}, and the time T_{send} to send the frame as a function of the data length Ld and transmission baudrate br.

$f_1()$ and $f_2()$ must be experimentally determined, as done in Section 4.

The total energy E_{tot} used for a transmission is therefore:

$$E_{\text{tot}} = E_{\text{radio}} + E_{\text{compression}}. \tag{3}$$

In the case where no compression is necessary the $E_{\text{compression}}$ is equal to zero.

3 Adaptive Near Sensor Compressing

If the operating conditions (location, data type, and frame length) of nodes are fixed and fully known at design stage, the best compressor can be fixed at design stage. However, in some cases (e.g. for moving or multi-sensor nodes), the length and the type of data to be transmitted can evolve during operation. In these cases, an adaptive system can select in real time the best compression scheme.

3.1 Data type and size influences on compression efficiency

In the rest of this paper, three different schemes applied to the WBAN data are being investigated: a Huffman compressor, a Bitpacking compressor and an additionally no compression scheme.

The Huffman coding replaces a word with a code whose length depends on the entropy of the data to compress. Bitpacking compression computes the dynamics of a data packet to remove the unnecessary heading and leading zeros or ones (present because default data storage format is

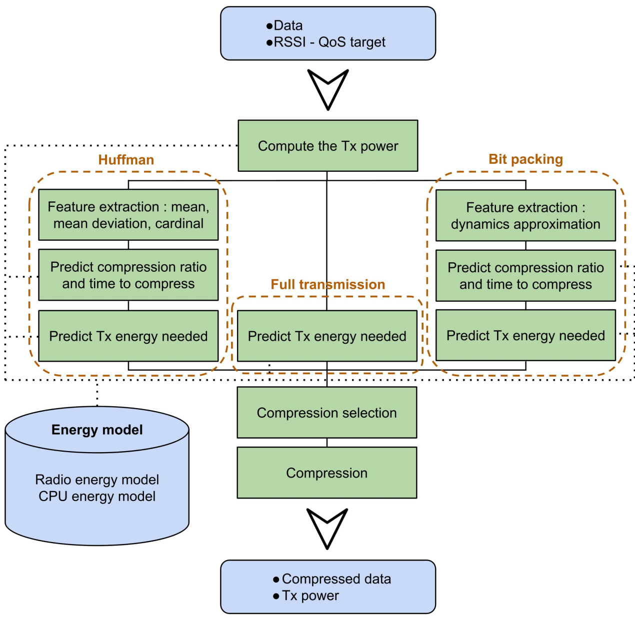


Figure 2. ANSC overview.

byte multiples). It is a simple compressor and thus it is very fast and does not need much memory, which makes it particularly suitable to WBAN.

Figure 1 shows maps of which is the most energy efficient scheme with respect to data length *Ld* and Tx power P_{Tx}. For each configuration (at a given couple *Ld* and P_{Tx}), the three schemes have been executed on a real node and the energy involved has been reported (considering both computation and transmission costs). Then the less consuming scheme is indicated by its respective color on the map. Two datasets issued from a real hardware WBAN platform (see Section 4) are considered: Figure 1a for data with high variance (`Zyggie moving` dataset) and Figure 1b for data with low variance (`Zyggie idle` dataset).

With the used compressors, a high variance dataset is less compressible than a low variance one. For small data length and low P_{Tx}, the compressors are not able to reduce the data enough to offset their own compression costs. However, for larger data and greater transmission power, compression can improve the total energy consumption. An adaptive system is therefore necessary to select the best compressor according to the data and transmission power.

3.2 Energy efficient compressor selection

To be efficient, ANSC must analyze the data and select the best scheme using the least energy as possible. As described in Figure 2, some pertinent feature extractions (their choice will be explained later in Section 4.2) are used to quickly estimate the compression ratio and, with the hardware and software models previously generated, the total energy consumption. The selected compression scheme is simply the less consuming one. Since energy estimations are done for each packet sent, an adaptation of the emission parameters and the choice of compression schemes is permanently carried out, allowing a high adaptability. The whole data analysis is generated through a single run to limit the memory access and thus reduce the fetch time involved.

4 Experimental results

4.1 WBAN Node architecture

The ANSC experiment has been performed on the WBAN Zyggie node developed at IRISA [5]. A Zyggie node embeds both a 32-bit ARM Cortex-M4 microcontroller which serves as the main controller and manages the different subsystems, and an ATMEGA128RF which contains an 8-bit microcontroller and a 2.4 GHz transceiver (802.15.4 compliant). The main sensor of a Zyggie node is an Inertial Measurement Unit (IMU) which delivers the spatial orientation of the node. This sensor is interesting in terms of compression, because when a node is in a resting place (i.e. when no movement occurs) the IMU data are nearly constant and on the other hand when movement occurs, the data have a high variance. This variation induces a shift in the various compression behaviors and can make a compressor efficient or not depending on the operation case (i.e. with the same P_{Tx} and *Ld*).

4.2 Model generation and feature extraction

To be able to select the right scheme on a specific architecture, ANSC uses combines different models and feature extractions in order to predict the compressor behaviors for each different data. The hardware and software models have been created using pertinent microbenchmarks and interpolations on real WBAN data. At this stage, only an estimation of the compression behaviour is made.

First, the radio model is obtained by estimating the functions $f_1()$ and $f_2()$ defined in (2). Polynomial functions are used to predict P_{send} and T_{send}, respectively:

$$E_{\mathrm{radio}} = (\gamma_0 + \gamma_1 * P_{\mathrm{Tx}} + \gamma_2 * {P_{\mathrm{Tx}}}^2 + \gamma_3 * {P_{\mathrm{Tx}}}^3 + \gamma_4 * {P_{\mathrm{Tx}}}^4) \cdot (\delta_0 + \delta_1 * Ld). \quad (4)$$

Table 1 gives the constant values of the model obtained by curve fitting on microbenchmarks for a given *br* of 250 kbps.

Figure 3 and Figure 4 show both measured and estimated values of P_{send} and T_{send}, respectively and prove the accuracy of the models.

The Bitpacking feature extraction consists of a fast estimation of the dynamics C_w of data packets. To this aim, a quick analysis of both ends (the first byte and the last ones, or only the first ones if the packet length is small) is performed. Along with C_w, the data length *Ld* is used to compute the

Table 1. Constant values of radio model.

γ_0	36.1		
γ_1	2.28	δ_0	0.448
γ_2	0.188		
γ_3	0.00902	δ_1	0.0328
γ_4	0.000183		

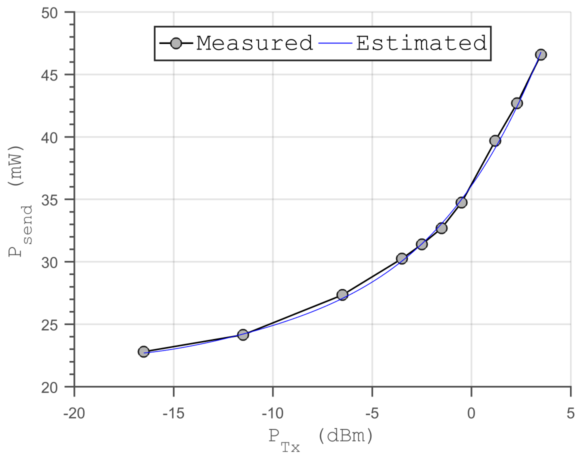


Figure 3. Power consumption as function of the P_{Tx}.

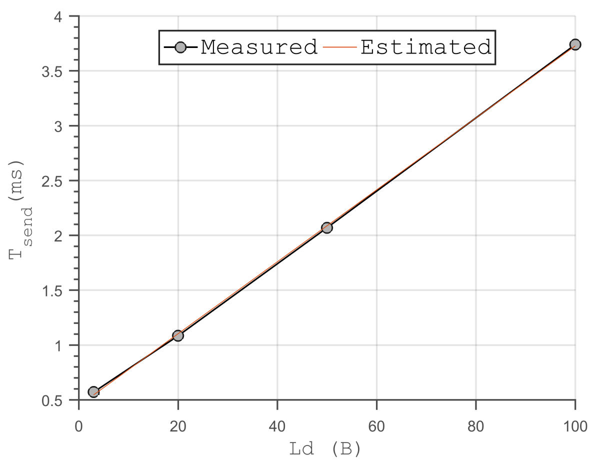


Figure 4. Time needed to send data frame as function of the frame length Ld at 250 kbps.

byte length at the output of the compressor $Lc_{\mathrm{Bitpacking}}$:

$$Lc_{\mathrm{Bitpacking}} = Ld \cdot \frac{C_w}{\alpha_1}. \quad (5)$$

The time resulting from this compression is also linear:

$$Tc_{\mathrm{Bitpacking}} = Ld \cdot C_w \cdot \alpha_2. \quad (6)$$

Table 2 gives the values of the model constants obtained by curve fitting on microbenchmarks.

Huffman prediction is more complex and requires feature extraction such as mean deviation and cardinal. Moreover, the curve fitting method used a symmetrical sigmoid function. This kind of curve allows a more accurate curve fitting than the standard polynomial curve which is used for Bitpacking estimation at the expense of extra computation time. The compression capability can be approximated with the data length to be compressed Ld, the cardinal of these data ($Card$), the mean deviation $meandev$ and computed constants (Table 3):

Table 2. Constant values of Bitpacking model.

α_1	8.68
α_2	0.6024

Table 3. Constant values of Huffman model.

β_1	0.2706263	β_7	0.321046
β_2	75.6904	β_8	-0.1384
β_3	0.001042757	β_9	530.4646
β_4	0.3898518	β_{10}	1.174898
β_5	0.9		
β_6	0.8		

$$Lc_{\mathrm{Huffman}} = \frac{Ld}{\beta_1 + \frac{\beta_2}{1+\left(\frac{Ld}{\beta_3}\right)^{\beta_4}}} \cdot \frac{meandev^{\beta_5}}{Card^{\beta_6}}. \quad (7)$$

Finally, the computation time is defined according to the compression ratio $\frac{Ld}{Lc_{\mathrm{Huffman}}}$ and the size of the data Ld:

$$Tc_{\mathrm{Huffman}} = \frac{Ld}{\beta_7 + \frac{\beta_8}{1+\left(\frac{Ld}{\beta_9}\right)^{\beta_{10}}} \cdot \frac{Ld}{Lc_{\mathrm{Huffman+1}}}}. \quad (8)$$

It should be kept in mind that these estimations are dependent both on the hardware architecture used and also on the software implementation. While not perfectly accurate, these estimations are sufficient to be able to choose the most efficient scheme.

4.3 Real measurements on WBAN platform

Four transmission schemes have been implemented in our Zyggie platform: Huffman and Bitpacking compressors, ANSC and full transmission (no compression without P_{Tx} tuned). For the evaluation fairness, the data used to evaluate the algorithms are different from those used to create the various models.

Figure 5 shows the total energy consumption E_{tot} considering the cost of transmission and compression with a fixed transmission power P_{Tx} (at -16.5 dBm) for different data lengths Ld. As shown by the figure, compression is not useful for a small amount of data and it is better to directly send the data without any modification. The energy overhead of ANSC is counterbalanced by its adaptivity. It chooses full transmission scheme for small Ld and switches to Huffman (at 50 B) then Bitpacking (at 200 B) when the compression leads to a lower total energy consumption (Figure 1b has been issued with the same kind of data). In this configuration, the maximum energy reduction is about 49% for long data size compared to full transmission.

Figure 6 shows the adaptability with respect to the distance between the nodes (i.e. making the P_{Tx} varying to respect a target QoS). Three different datasets have been used: one of room temperature monitoring (`Temperature`

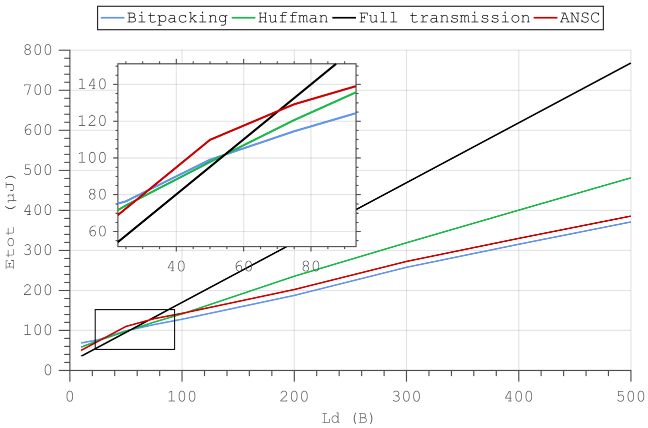


Figure 5. Total energy at fixed $P_{\mathbf{Tx}}$ (-16.5 dBm) for different Ld.

data) and two issued from our Zyggie WBAN node (`Zyggie moving` and `Zyggie idle` sets). These later are issued from the same sensor but have different statistics (according to their operating case), the `Zyggie idle` set has a low variance whereas the `Zyggie moving` one has a higher variance.

Since the data size is the same for the three datasets, the three curves of full transmission are exactly overlapped. The two Zyggie datasets have a different compression capability. The `Zyggie moving` set is less compressible (more bits will be sent by radio) than the two other sets and the blue curves shows a higher consumption E_{tot} than for the others datasets. In the worst case, the minimum energy reduction acheieved by ANSC with the regard to full transmission scheme is about 23% on `Zyggie moving` data and the best case is of 45% on a $210\,B$ `Zyggie idle` data frame. The best algorithm for `Zyggie idle` set is Bitpacking, whereas Huffman is efficient for the `Zyggie moving` one, and this well handled by ANSC. Globally the change of Tx power has less influence on the scheme choice than the data type.

5 Conclusion

In this paper, several compression techniques have been tested on different WBAN datasets in order to evaluate their capacity to reduce the radio energy consumption. In most cases, there does not exist a superior compression technique relevant for any WBAN dataset and we show that the choice of a technique depends on the data types, the amount of data and the distance between nodes. A solution is to dynamically switch to the best technique to improve the energy reduction. The proposed adaptive algorithm (ANSC) chooses in real time the most energy efficient technique based on prediction model and also dynamically adapts the transmission power to further reduce the radio consumption. ANSC has been validated in a real use case on a WBAN node. An energy saving up to 49% has been reached for this WBAN platform. As this result is very dependent on the node hardware, this energy saving should be even higher with a lower power CPU or a more consuming radio chip such as LoRa standard.

6 References

[1] MiniLZO website : http://www.oberhumer.com/opensource/lzo/.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless Sensor Networks: A Survey. *Computer Network*, 38(4):393–422, march 2002.

[3] M. M. Alam, O. Berder, D. Menard, and O. Sentieys. On the Energy Savings of Adaptive Transmit Power for Wireless Sensor Networks Radio Transceivers. In *Proc. International Conference on Architecture of Computing Systems (ARCS)*, volume 1, pages 1–6, Feb 2013.

[4] J. Chang and L. Tassiulas. Energy conserving routing in wireless adhoc networks. In *Proc. IEEE International Conference on Computer Communications (INFOCOM)*, volume 1, pages 22–31, march 2000.

[5] A. Courtay, M. Le Gentil, O. Berder, A. Carer, P. Scalart, and O. Sentieys. Zyggie: A Wireless Body Area Network platform for indoor positioning and motion tracking. In *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–5, May 2018.

[6] X. Deng and Y. Yang. Online Adaptive Compression in Delay Sensitive Wireless Sensor Networks. *IEEE Transactions on Computers*, 61(10):1429–1442, Oct 2012.

[7] J. Haapola, Z. Shelby, C. Pomalaza-Raez, and P. Mahonen. Crosslayer energy analysis of multi-hop wireless sensor networks. In *Proc. of the Second European Workshop on Wireless Sensor Networks (EWSN)*, 2005.

[8] D. Huffman. A method for the construction of minimum-redundancy codes. *I.R.E.*, pages 1098–1102, 1952.

[9] J. Kim, J. Kim, and C. Kyung. A lossless embedded compression algorithm for high definition video coding. In *IEEE International Conference on Multimedia and Expo*, pages 193–196, June 2009.

[10] J. Reason and J. Rabaey. A Study of Energy Consumption and Reliability in a Multi-hop Sensor Network. *SIGMOBILE*, 8(1):84–97, jan 2004.

[11] C. Sadler and M. Martonosi. Data Compression Algorithms for Energy-constrained Devices in Delay Tolerant Networks. In *Proc. International Conference on Embedded Networked Sensor Systems (SenSys)*, pages 265–278, Oct 2006.

[12] A. Van Der Byl, R. Neilson, and R. H. Wilkinson. An evaluation of compression techniques for Wireless Sensor Networks. In *IEEE AFRICON 2009*, pages 1–6, Sep. 2009.

[13] B. Ying. An adaptive compression algorithm for energy-efficient wireless sensor networks. In *Proc. International Conference on Advanced Communication Technology (ICACT)*, pages 861–868, Feb 2017.

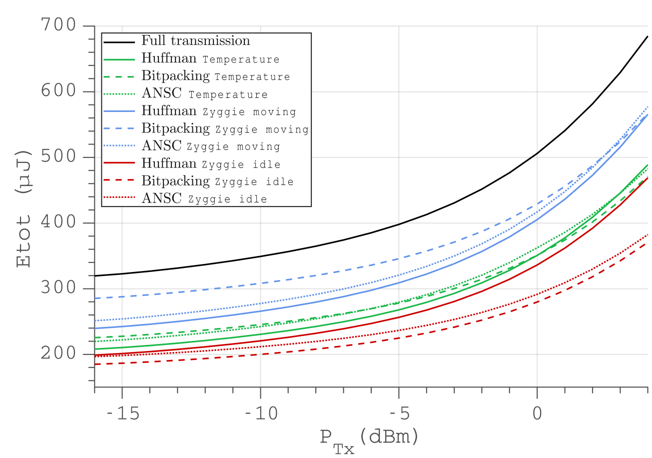


Figure 6. Total energy at fixed Ld ($210\,B$) for different $P_{\mathbf{Tx}}$ and different WBAN datasets.